

Citation for published version:

Ustyuzhaninov, I, Kazlauskaitė, I, Kaiser, M, Bodin, E, Campbell, N & Ek, CH 2019, 'Compositional Uncertainty in Deep Gaussian Processes', Paper presented at The Conference on Uncertainty in Artificial Intelligence, 3/08/20 - 6/08/20.

Publication date:
2019

[Link to publication](#)

University of Bath

Alternative formats

If you require this document in an alternative format, please contact:
openaccess@bath.ac.uk

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Compositional uncertainty in deep Gaussian processes

Ivan Ustyuzhaninov*
University of Tübingen

Ieva Kazlauskaitė*
University of Bath,
Electronic Arts

Markus Kaiser
Siemens AG,
TU Munich

Erik Bodin
University of Bristol

Neill D. F. Campbell
University of Bath,
Royal Society

Carl Henrik Ek
University of Bristol

Abstract

Gaussian processes (GPs) are non-parametric priors over functions, and fitting a GP to the data implies computing the posterior distribution of the functions consistent with the observed data. Similarly, deep Gaussian processes (DGPs) [1] should allow us to compute the posterior distribution of compositions of multiple functions giving rise to the observations. However, exact Bayesian inference is usually intractable for DGPs, motivating the use of various approximations. We show that the simplifying assumptions for the mean field variational approximation imply that all but one layer of a DGP collapse to a deterministic transformation. We argue that such an inference scheme is suboptimal, not taking advantage of the potential of the model to discover the compositional structure in the data, and propose possible modifications addressing this issue.

1 Introduction

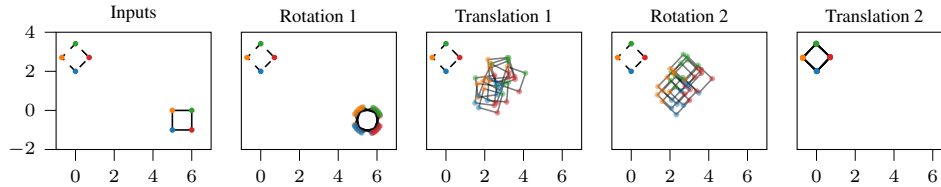


Figure 1: Example of a compositional model: transformation of a solid rectangle onto the dashed one is decomposed as $T_2 \circ R_2 \circ T_1 \circ R_1$ with R_i and T_i being rotations and translations. Different sampled realisations of these transformations are overlaid, showing the *compositional uncertainty*. Regular DNNs and DGPs do not capture such uncertainty, collapsing to a single realisation of a composition to model the data.

Deep learning studies functions represented as compositions of other functions, $f = f_L \circ \dots \circ f_1$. Compositions of functions is a natural way to model data generated by a hierarchical process. Each f_i represents a certain part of the hierarchy, and the prior assumptions on f_i reflect the corresponding prior assumptions about the data generating process. Moreover, if the data generating process is not hierarchical, compositional models offer a convenient way to impose priors on simpler f_i rather than on the overall complex function, which in practice might result in a model that is easier to fit. Given these prior assumptions, we can compute the posterior distributions of f_i and by doing so uncover the structure of the data and explicitly estimate the uncertainties due to each function (or layer) in the composition.

* Equal contribution

The uncertainties in each f_i give rise to what we call compositional uncertainty: even noiseless observed data could be generated by compositions of a multitude of different functions which are consistent with the prior (for example, see Fig. 1). An example of a problem where computing such uncertainty is important is alignment of temporal signals [5, 6], where it is informative to not only compute a point estimate of the temporal warps aligning the signals to each other, but also see the range and likelihood of different possible warps. Another example is transfer learning where we assume that two hierarchical models $m_1 = g_1 \circ f$ and $m_2 = g_2 \circ f$ share a common part of the hierarchy (f in this example). Having fitted m_1 , we can fit m_2 to a different data set (or domain) by reusing f from m_1 and fitting only g_2 . In this case it is important to capture a wide distribution of possible realisations of f , such that it adequately models a common part of m_1 and m_2 (as opposed to finding a single realisation of f , only useful for explaining the data set we used for fitting m_1).

The research in deep neural networks (DNNs), however, has mostly focused on networks built using a very large number of simple functions where the goal is to fit the entire composition to the data [2], while the computations by individual functions in the composition or parts of the network are often irrelevant and not interpretable. In other words, DNNs generally do not model a hierarchical process but only the (predictive) distribution of the data. The model is thus much closer to a single-layer GP and the issue of compositional uncertainties are ignored altogether.

DGPs [1], which are compositions of GPs, allow us to impose explicit prior assumptions on f_i by choosing the kernels, and perform Bayesian inference to compute the posterior over each layer that is consistent with the observed data. Such posteriors would capture the compositional uncertainty showing the range of transformations fitting the data. We note that DGPs are inherently unidentifiable, since different compositions can fit the data equally well, and we argue that it should be captured by an adequate Bayesian posterior. However, exact Bayesian inference in DGPs is intractable [1] and we have to resort to approximations. We show that typically used approximate inference schemes make strong simplifying assumptions resulting in all but one of the layers of a DGP collapsing to deterministic transformations. In certain cases (e.g. when we have weak priors on the layers or uncertainty is irrelevant for the application) such behaviour is not an issue, however, in general that prevents us from using the power of probabilistic models to capture the uncertainty in the hierarchy. By highlighting these limitations of the current inference schemes and suggesting their modifications, we aim to show that the assumptions on the approximate inference scheme are central to the estimation of compositional uncertainty.

2 Issues with compositional uncertainty in DGPs

DGPs are compositions of functions $f = f_L \circ \dots \circ f_1$, where $f_i(\cdot) \sim \mathcal{GP}(\mu_i, k_i(\cdot, \cdot))$. Having observed the data set $(X, Y) = \{(x_i, y_i)\}_{i=1}^N$, the marginal likelihood

$$p(Y | X) = \int \dots \int p(Y | F_L) p(F_L | F_{L-1}) \dots p(F_1 | X) dF_L \dots dF_1, \quad (1)$$

where $p(F_i | F_{i-1}) \sim f_i(F_{i-1})$

is intractable since it requires integration of the non-linear covariance matrices in the terms $p(F_i | F_{i-1})$. As proposed by [1, 8], a lower bound on this intractable integral can be estimated using variational approximations based on augmenting the GPs f_i with inducing points U_i , which are treated as variational parameters. Conditioned on the inducing points, the output distribution of each layer can be computed as a GP posterior distribution, treating U_i as (pseudo-) observations. Introducing a variational distribution $q(U_1, \dots, U_L)$, the marginal likelihood lower bound is computed as (see [8] for further details):

$$p(Y | X) \geq \mathbb{E}_{q(F_L)} [\log p(Y | F_L)] - \text{KL}[q(U_1, \dots, U_L) || p(U_1) \cdot \dots \cdot p(U_L)]. \quad (2)$$

2.1 Collapse of intermediate layers to deterministic transformations

The variational distribution over the inducing points is typically chosen to be factorised as $q(U_1, \dots, U_L) = q(U_1) \cdot \dots \cdot q(U_L)$, however, that leads to the layers of a DGP collapsing to deterministic transformations [4]. As demonstrated in Figs. 2 (first row) and 3, different random samples from a DGP fitting the same data look essentially the same, meaning there is almost no uncertainty captured about the transformations in intermediate layer. At the same time, fitted DGPs

corresponding to different random initialisations (denoted by different colours in the figures) converge to different solutions, indicating that there are multiple compositions of these three functions fitting the data, which should be captured as part of the compositional uncertainty.

We argue that this uncertainty collapse is due to the factorisation of the variational distribution over the inducing points. The inducing points essentially define the transformation in each layer (more precisely, the output distribution of each layer $P(F_i | F_{i-1}, U_i)$ is parametrised by the inducing points). Hence the variational distribution $q(U_i)$ induces a distribution over the mappings implemented by that layer, but these transformations in different layers are independent. However, in order to fit the observed data, the layers must be dependent (e.g. if f_1 implements a blue transformation in Fig. 2, f_2 must implement a blue transformation as well, otherwise the entire composition does not fit the data), and that the only way to achieve that with a factorised variational distribution is to make each factor essentially a point mass on some transformation. This is akin to subsequent layers having noisy inputs as in [3], which leads to increased output uncertainty. Another illustration of this idea is provided in Fig. 1, where we fitted a model with correlated rotations and translations, allowing us to see the variety of possible motions of the square. However, a model with independent transformations in each layer would converge to a single possible sequence of rotations and translations.

3 Modeling dependencies between layers

We discuss and compare two ways of introducing dependencies between the inducing points in order to capture the compositional uncertainty.

Jointly Gaussian inducing points A straightforward way to introduce dependencies between the inducing points is to define $q(U_1, \dots, U_L) \sim \mathcal{N}(\mu, \Sigma)$ with a joint covariance matrix across layers. In this case, the expectation in Eq. (2) can be approximated numerically by drawing a sample from $q(U_1, \dots, U_L)$, then, conditioned on the sampled U_i , drawing the DGP output sample by sequentially sampling from GP posterior distributions in each layer (in the same way as in [8]), and finally using these sampled DGP outputs to compute a Monte-Carlo estimate of the expectation. The reparametrisation trick [7] for the Gaussian distribution permits computation of the gradients of the variational parameters.

Variational distributions of outputs of intermediate layers Modelling the correlations between the inducing points directly does not scale with the number of inducing points and with the depth of a DGP. To address these issues we propose the following variational distribution and inference scheme.

We introduce the inducing locations Z in the input space, and the outputs of intermediate layers $\{F_i^Z\}$ at these inducing inputs: $F_i^Z \sim (f_i \circ \dots \circ f_1)(Z)$ with $F_0^Z = Z$. Denoting the outputs of the intermediate layers at observed inputs as $\{F_i\}$, i.e. $F_i \sim (f_i \circ \dots \circ f_1)(X)$, $F_0 = X$ and $F_L = Y$, the DGP joint distribution is as follows:

$$p(Y, F_L, \dots, F_1, F_L^Z, \dots, F_1^Z | X, Z) = p(Y | F_L) \prod_{i=1}^L p(F_i | F_{i-1}, F_i^Z, F_{i-1}^Z) p(F_i^Z | F_{i-1}^Z) \quad (3)$$

where the terms $p(F_i | F_{i-1}, F_i^Z, F_{i-1}^Z)$ are the GP posteriors given the outputs of the intermediate layers at the inducing inputs, and $p(F_i^Z | F_{i-1}^Z)$ are the GP priors in each layer.

We introduce the following variational distribution:

$$q(\{F_i\}, \{F_i^Z\}) = \prod_{i=1}^L p(F_i | F_{i-1}, F_i^Z, F_{i-1}^Z) q(F_i^Z), \quad (4)$$

with $q(F_i^Z) \sim \mathcal{N}(\mathbf{m}_i, \mathbf{S}_i)$ as free-form Gaussians. Using such variational distribution we obtain the likelihood lower bound:

$$\begin{aligned} \log p(Y) &\geq \mathbb{E}_q \left[\log \frac{p(Y, \{F_i\}, \{F_i^Z\} | X, Z)}{q(\{F_i\}, \{F_i^Z\})} \right] \\ &= \mathbb{E}_{q(F_L)} [\log p(Y | F_L)] \end{aligned} \quad (5)$$

$$- \sum_{i=1}^L \mathbb{E}_{q(F_{i-1}^Z) q(F_i^Z)} \left[\log \frac{q(F_i^Z)}{p(F_i^Z | F_{i-1}^Z)} \right]. \quad (6)$$

We estimate the expectation (5) by sampling from $q(F_L)$ using the following procedure:

1. Draw S samples $(F_i^Z)^1, \dots, (F_i^Z)^S \stackrel{iid}{\sim} q(F_i^Z)$ for $i = 1, \dots, L$.
2. For each of the S sampled values of the intermediate layers at the inducing inputs (i.e. $(F_1^Z)^s, \dots, (F_L^Z)^s$ for $s = 1, \dots, S$), draw $(F_L)^s \sim q(F_L)$ by sequentially drawing from $p(F_i | F_{i-1}, F_i^Z, F_{i-1}^Z)$ using the procedure described in [8].
3. Use the samples from $q(F_L)$ for the Monte-Carlo estimate of (5):

$$\mathbb{E}_{q(F_L)}[\log p(Y | F_L)] \approx \frac{1}{S} \sum_{s=1}^S \log p(Y | (F_L)^s).$$

The inner expectations in (6) can be written as KL divergences:

$$\mathbb{E}_{q(F_i^Z)q(F_{i-1}^Z)} \left[\log \frac{q(F_i^Z)}{p(F_i^Z | F_{i-1}^Z)} \right] = \mathbb{E}_{q(F_{i-1}^Z)} \text{KL}[q(F_i^Z) || p(F_i^Z | F_{i-1}^Z)]. \quad (7)$$

For a given value of F_{i-1}^Z , $\text{KL}[q(F_i^Z) || p(F_i^Z | F_{i-1}^Z)]$ can be computed analytically as a KL divergence between two Gaussians. That suggests the following procedure for estimating the summands in (6):

1. Draw S samples $(F_i^Z)^1, \dots, (F_i^Z)^S \stackrel{iid}{\sim} q(F_i^Z)$ for $i = 1, \dots, L$ (those can be the draws used for estimating (5) above).
2. For each of the S sampled values of F_{i-1}^Z , analytically compute $\text{KL}[q(F_i^Z) || p(F_i^Z | (F_{i-1}^Z)^s)]$.
3. Compute the Monte-Carlo estimate of (6):

$$\mathbb{E}_{q(F_i^Z)q(F_{i-1}^Z)} \left[\log \frac{q(F_i^Z)}{p(F_i^Z | F_{i-1}^Z)} \right] \approx \frac{1}{S} \sum_{s=1}^S \text{KL}[q(F_i^Z) || p(F_i^Z | (F_{i-1}^Z)^s)].$$

4 Experiments

Jointly Gaussian inducing points In Figs. 2 (second row) and 4 we show examples of fits of DGPs with jointly Gaussian inducing points. In comparison to Figs. 2 (first row) and 3, such an approach indeed retains more of the (compositional) posterior uncertainty about the transformations in the intermediate layers avoiding collapsing to point estimates. However, different initialisations still result in different posteriors, highlighting the limited capacity of the Gaussian distributions to capture the complex (e.g. multimodal) posteriors, as previously discussed in [4].

Variational distributions of outputs of intermediate layers Example fits of DGPs using the variational distribution (4) are shown in Figs. 2 (third row) and 5. The posteriors are qualitatively similar to those obtained using a jointly Gaussian variational distribution, with an advantage of the number of the parameters in the proposed distribution scaling linearly rather than quadratically with the number of layers. Such a variational distribution also cannot capture multimodal posteriors, requiring future work on more flexible variational distributions.

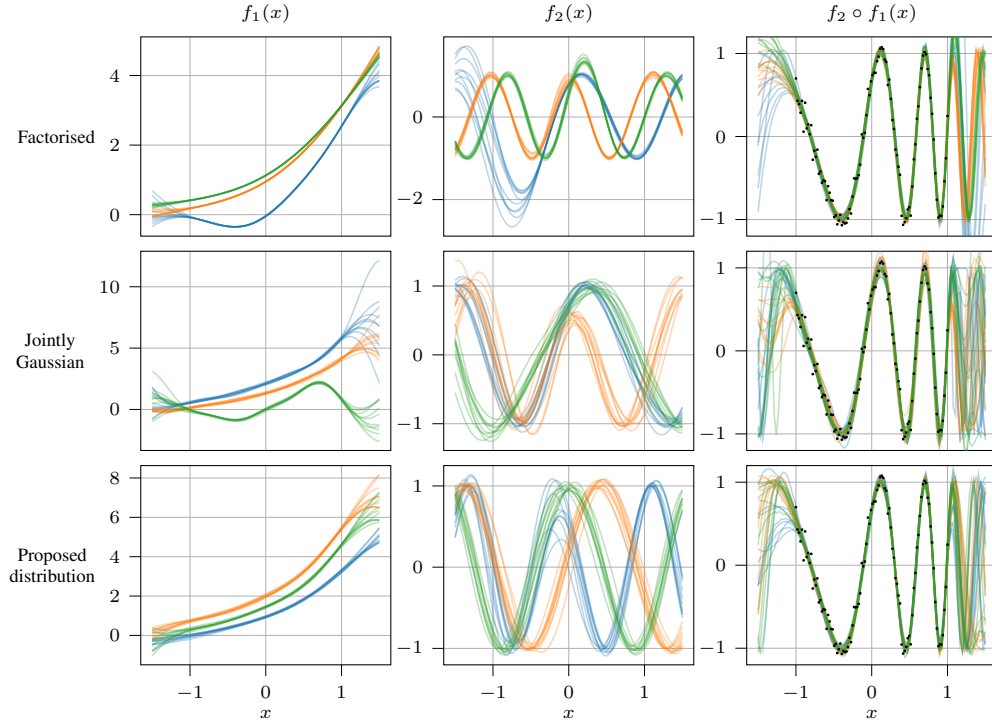


Figure 2: Example fits of a two-layer DGP with inducing points distributed according to a factorised, jointly Gaussian, or proposed variational distribution (Eq. (4)). Data set is shown in the rightmost panels (black dots). Different panels show the computations performed by each of the two layers and their compositions. The kernel in the first layer is squared-exponential, while in the second one constrained to be periodic. Different colours correspond to three models fitted to the same data with different initialisations. For each initialisation, ten samples (of the same colour) from the fitted model are shown on top of each other. For factorised inducing points, the transformations in the second layer collapse to deterministic ones in the output range of the first layer, while some uncertainty about such transformations is preserved for models with dependent inducing points across layers.

References

- [1] A. Damianou and N. Lawrence. Deep gaussian processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2013.
- [2] Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.
- [3] A. Girard, C. E. Rasmussen, J. Q. Candela, and R. Murray-Smith. Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.
- [4] M. Havasi, J. M. Hernández-Lobato, and J. J. Murillo-Fuentes. Inference in deep gaussian processes using stochastic gradient hamiltonian monte carlo. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [5] M. Kaiser, C. Otte, T. Runkler, and C. H. Ek. Bayesian alignments of warped multi-output gaussian processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [6] I. Kazlauskaitė, C. H. Ek, and N.F.D. Campbell. Gaussian process latent variable alignment learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2019.
- [7] D. P Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Representation Learning (ICLR)*, 2014.
- [8] H. Salimbeni and M. Deisenroth. Doubly stochastic variational inference for deep gaussian processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Additional figures

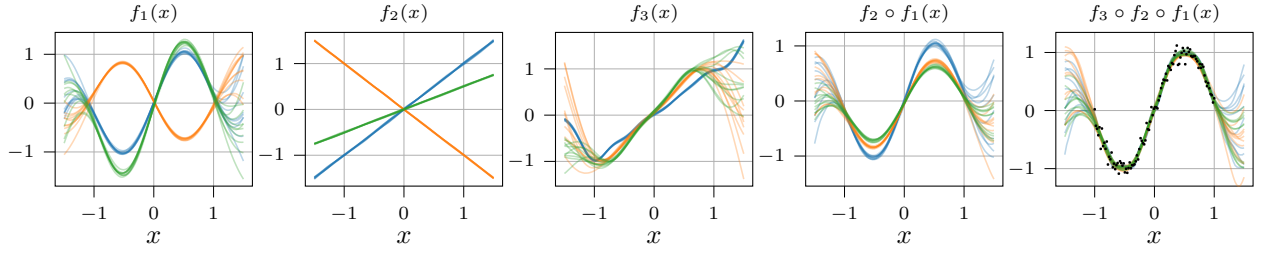


Figure 3: Example fits of a three-layer DGP with *factorised inducing points* to a data set shown in the rightmost panel (black dots). Different panels show the computations performed by each of the three layers and their compositions. Different colours correspond to three models fitted to the same data with different random initialisations. For each initialisation, ten samples (of the same colour) from the fitted model are shown on top of each other.

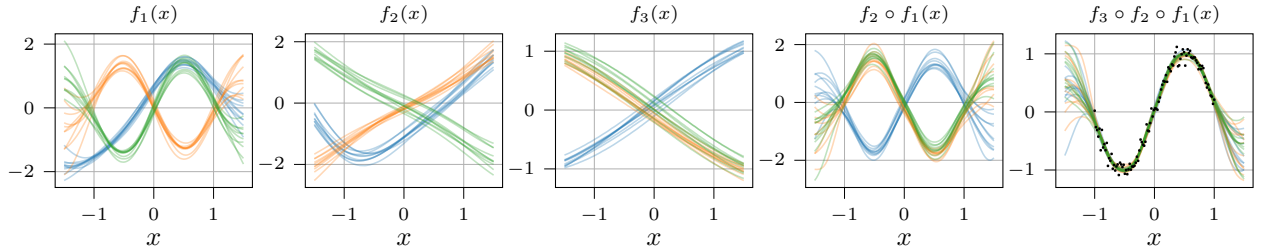


Figure 4: Example fits of a three-layer DGP with *jointly Gaussian inducing points* (Sec. 3). The figure arrangement is the same as in Fig. 3.

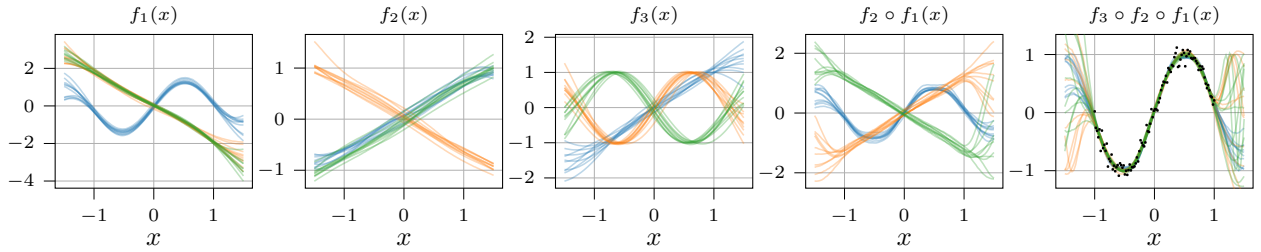


Figure 5: Example fits of a three-layer DGP with *variational distribution of outputs of intermediate layers* (Sec. 3). The figure arrangement is the same as in Fig. 3